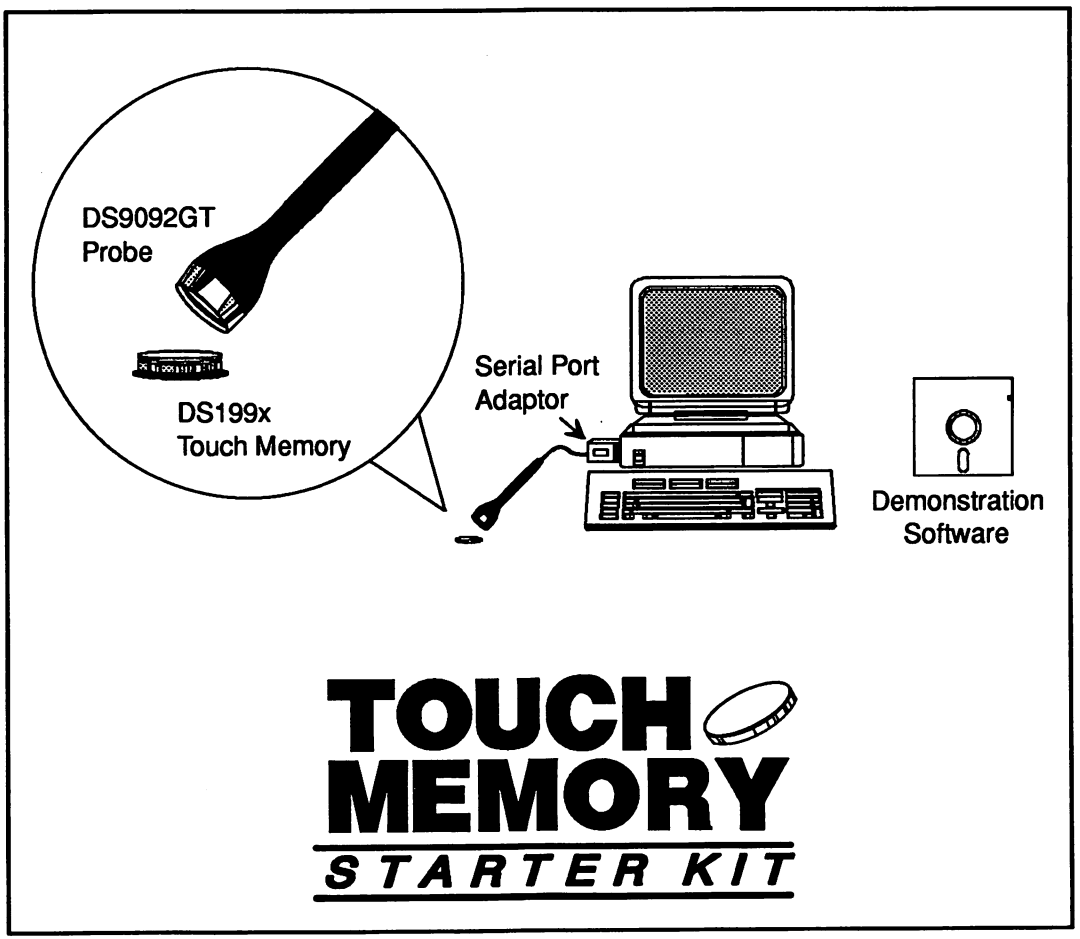
DALLAS SEMICONDUCTOR

DS9092K
Touch Memory Starter Kit



INTRODUCTION

The DS9092K Touch Memory Starter Kit provides a quick and simple method for evaluating Dallas Semiconductor's family of Touch products.

This kit contains the following:

Qty	Part Number	Description
5	DS199x	Touch Memories (Assorted)
1	DS9092GT	Touch Memory with Tactile Feedback and Grip
1	DS9092	Touch Memory Probe
1	DS9097	Touch Memory Serial Port Adaptor
1	DS0600-5	Touch Memory PC Software (2 Floppy Disks)
1	DS9093	Touch Memory FOB Mount
1	DS9093F	Touch Memory Flanged FOB Mount
1	DS9093S	Touch Memory Permanent Mount
1	DS9094F	Touch Memory Clip F5
12	DS9096	Touch Memory Adhesive Pad
1	DS9093RA	Touch Memory Lock Ring Fastener
1		Dallas Semiconductor Databook and other Documentation

TOOLS AND EQUIPMENT NEEDED

- IBM PC/XT/AT/EISA or compatible with 5-1/4" floppy drive and serial communications port that runs to 115.2 Kbps
- Medium flat blade screwdriver

GETTING STARTED

1. Connect the DS9092 Touch Memory Probe to the DS9097 PC Serial Port Adaptor. The probe has an RJ11 connector (telephone jack connector) on the end of the wire. This plug snaps into the mating connector on the PC adaptor.
2. Plug the 25-pin connector of the adaptor into the serial port of the computer and securely fasten it with the captive screws. If your computer has a 9-pin serial port, use a standard 25-pin to 9-pin adaptor to complete the connection. (Be sure the adaptor is a straight-through type and not a null modem or 2-3 swap.) See Figure 1.
3. Insert the diskette into drive A.
4. Review the README file before beginning the demo by typing:

   A: README <*ENTER*>

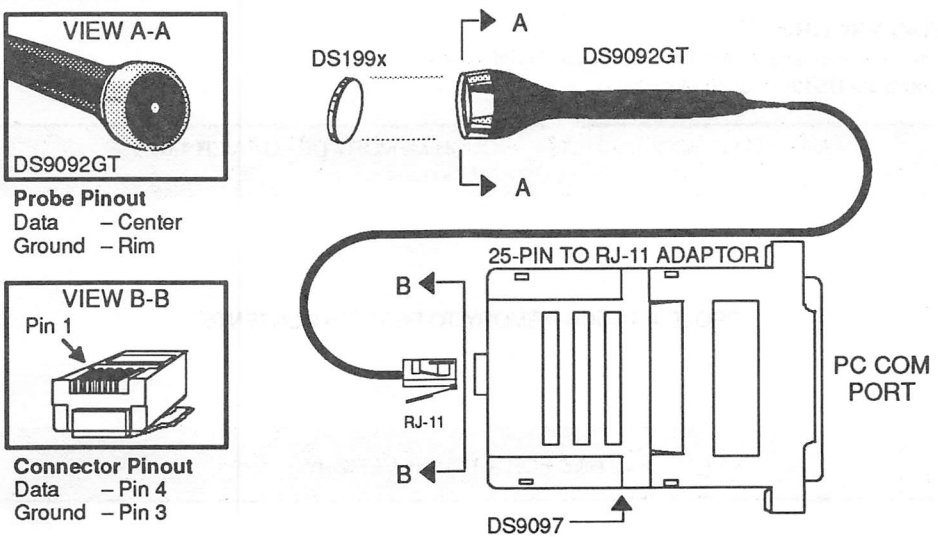
5. To start the program, type:

   A: TOUCH <*ENTER*>

If you are using a COM port other than 1, add the port number. For example, if you are using COM port 2, type:

A: TOUCH 2 <*ENTER*>

TOUCH MEMORY – PC READER/WRITER ASSEMBLY Figure 1



Once the program is underway, a menu and instructions will appear on the screen as follows:

MAIN MENU

Dallas Semiconductor Touch Memory
DS199x Demonstration
Version 2.0 Copyright (C) 1992

1) Introduction to the DS199x Demo
2) LESSON 1: UNIQUE ROM ID NUMBER
3) LESSON 2: READING A PAGE
4) LESSON 3: USING THE SCRATCHPAD
5) LESSON 4: FORMATTING DATA
6) LESSON 5: DS1994 CLOCK FUNCTIONS
7) DS199x READ/WRITE UTILITY

<Arrow> Select <Enter> Accept <ESC> Quit

MAIN ITEM DESCRIPTION

The Introduction describes the different parts of the demonstration and the informal windows used in these parts.

This demo program will work with the DS1990 Touch Serial Number, DS1992 Touch Memory 1K, DS1993 Touch Memory 4K, and the DS1994 Touch Time and Memory. To evaluate the DS1991 Touch MultiKey, the 91_Demo Program is used. The instructions for this demo are in the next section of this manual.

To exit the Touch program, press <ESC> and you will return to the DOS prompt.

91_DEMO PROGRAM

This program demonstrates the DS1990 Touch Serial Number and the DS1991 Touch MultiKey.

To start the program, type:

A: 91_Demo <ENTER>

If you are using a COM port other than 1, add the port number.

Once the program is underway, the following screen will appear:

DALLAS SEMICONDUCTOR TOUCH MEMORY DEMO VER 1.00
QUIT <ALT>+Q

PROBE A TOUCH MEMORY TO READ ITS CONTENTS

WAITING FOR A TOUCH MEMORY

If the screen above does not appear, the com port adaptor may not be properly installed or may be on the wrong COM port. In the TOOLBOX subdirectory on the applications disk is a hardware diagnostic program, DIAGNOSE.EXE. To execute DIAGNOSE, enter DIAGNOSE 1 to test COM port 1 or DIAGNOSE 2 to test COM port 2.

If the screen does appear, touch a DS1990 touch serial number to the Touch Memory Probe. The following screen will appear:

```
DALLAS SEMICONDUCTOR   TOUCH MEMORY DEMO   VER 1.00
QUIT <ALT>+Q

                    ROM DATA

                    FAMILY CODE     : 01
                    SERIAL NUMBER   : 00 00 00 04 F4 97
                    CRC             : BE

*DS1990          – 64–BIT ROM
*FAMILY CODE     – TOUCH MEMORY MODEL NUMBER
*SERIAL NUMBER – UNIQUE REGISTERED LASER NUMBER WRITTEN AT TIME OF
                   MANUFACTURE
*CYCLICAL REDUNDANCY CHECK – CHECK CODE TO VERIFY DATA

                 DS1990 TOUCH MEMORY FOUND
```

The data displayed includes the family code (01 for a DS1990), the unique serial number, and the CRC.

READING A DS1991 TOUCH MULTIKEY

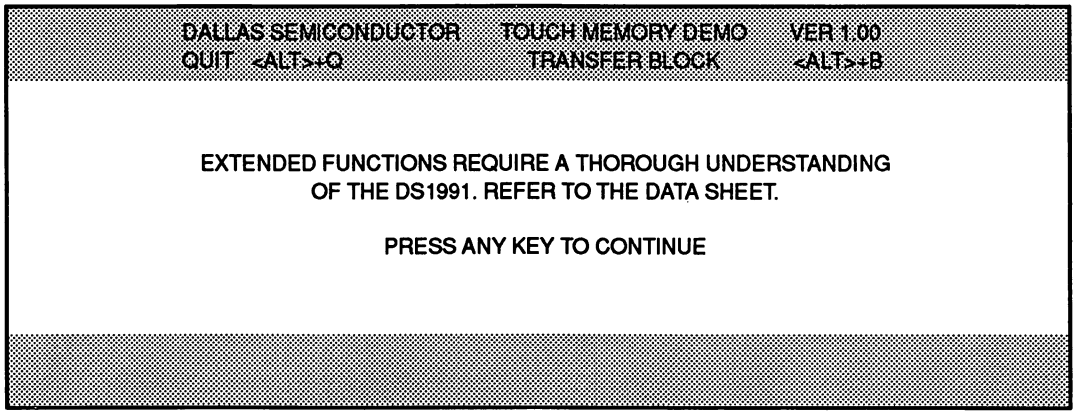
To read or write a DS1991, start the program as discussed earlier. Probe a DS1991. The following screen will appear:

```
DALLAS SEMICONDUCTOR   TOUCH MEMORY DEMO   VER 1.00
QUIT <ALT>+Q           EXTENDED FUNCTIONS  <ALT>+E

                    ROM DATA

                FAMILY CODE     : 02

                SERIAL NUMBER   : 00 00 00 01 D2 DA

                CRC             : 23

                READ/WRITE MEMORY
[Print your own message here, press <enter>, then probe a DS1991]

TO ACCESS ALL 2048-BIT & EXTENDED FUNCTIONS, PRESS <ALT>+E

TO READ JUST PROBE.  TO WRITE TYPE DATA, PRESS <ENTER>, THEN PROBE

                 DS1991 TOUCH MEMORY FOUND
```

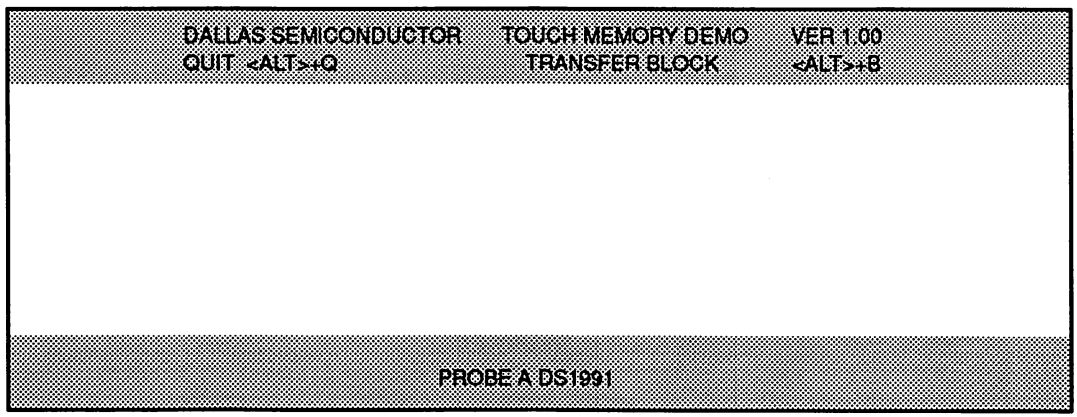
You will notice the family code, the unique serial number, and the CRC value, just like the DS1990. The bracketed area contains the contents of the DS1991 scratchpad memory. To write new information into the scratchpad memory, use the arrow keys to move the cursor down to the bracketed area, type in your message, press <enter>, then probe a DS1991.

DS1991 TUTORIAL – EXTENDED FUNCTIONS

At this point, you may want to experiment with the extended functions of the DS1991 touch multikey. Enter <ALT>+E. The following screen will appear.



Press <enter>. The following screen will appear:



At this point, the program is waiting to probe a DS1991. Probe a DS1991 and hold it until the following screen is displayed:

```
DALLAS SEMICONDUCTOR    TOUCH MEMORY DEMO    VER 1.00
QUIT <ALT>+Q                TRANSFER BLOCK       <ALT>+B

                    ROM DATA: [ 51  00 00 00 00 4A 1F  02 ]

                    SCRATCHPAD

          [This section will contain the contents of the scratchpad.]

                    KEY NUMBER 0

PASSWORD:       [           ]
ID FIELD:       [OLD ID 0 ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    KEY NUMBER 1

PASSWORD:       [           ]
ID FIELD:       [OLD ID 1 ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    KEY NUMBER 2

PASSWORD:       [           ]
ID FIELD:       [OLD ID 2 ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    PROBE A DS1991
```

The contents of the DS1991 scratchpad which you just probed will be displayed in the top brackets. Notice that the PASSWORD field of each subkey is blank – you can't read a part's password; you can only write it. For the purposes of the demo program, once you write a password in the password field, that value will remain displayed and be used for access to the part until it is changed. At the device level, the password field is write only. The ID FIELDs of each of the three subkeys are the ID fields read from the part. The SECURE DATA field will display random data – the real information can't be read unless the correct password for a given subkey is known and entered.

Before we continue, a brief overview of the DS1991 is required.

The DS1991 Touch MultiKey is a 2K–bit NV memory that is divided into four sections:

Scatchpad	512 bits
SubKey 0	64–bit ID
	64–bit Password
	384–bit Secure Memory

SubKey 1	64–bit ID
	64–bit Password
	384–bit Secure Memory
SubKey 2	64–bit ID
	64–bit Password
	384–bit Secure Memory

The EXTENDED FUNCTIONS screen in the Touch Demo provides complete information on all of the areas in the DS1991 at the same time. Using the four <arrow> keys to position the cursor, data can be entered into any of these fields.

A Touch Memory is read every time it is probed, and all of the updated information is displayed on the screen.

Writing to a Touch Memory is quite simple, but slightly different for each of the different types of fields such as the scratchpad, subkey 0, etc.

SCRATCHPAD

This section of read/write memory serves as a temporary location to store incoming data and then verify it before transferring it to one of the secure subkeys.

To enter data into the scratchpad, simply key in the desired data, press <ENTER>, and then probe the DS1991. The data is buffered up in the computer until <ENTER> is pressed. Data is then transferred from the computer to the Touch Memory when probed.

SUBKEYS

Each subkey is made up of three parts: the ID Field, the Password and the Secure Data. Access to the secure data requires a correct match to the password stored in the DS1991. When the password is correctly entered, the secure data of that specific subkey can be read or modified. **Data should be written to the scratchpad, verified, and then transferred to the specified subkey.** Although data can be read and written directly to and from a subkey, it is not suggested.

The ID field for each subkey is like a label for that subkey. For instance, Subkey0 might be to the house, where Subkey1 is to the car and Subkey2 is to the office. The ID field is 64 bits (8 bytes) long. Its contents are NOT secure and can be read by anyone. In the demo program, the ID field is used to differentiate between entering a password match and setting a new password. "Set password" is a command of the DS1991 which is separate from submitting a password to gain access to the secure data.

Whenever a password is set in the DS1991, the secure memory is cleared. So, in this demo program, whenever the ID field is changed, the value in the password field is used as the new valid data to set the new password. Therefore, the secure data is cleared.

A 64–bit (8–byte) password is required to access each of the subkeys. Each subkey's password is independent and can be set to any value.

ACCESSING SUBKEYS

1) Setting the Password

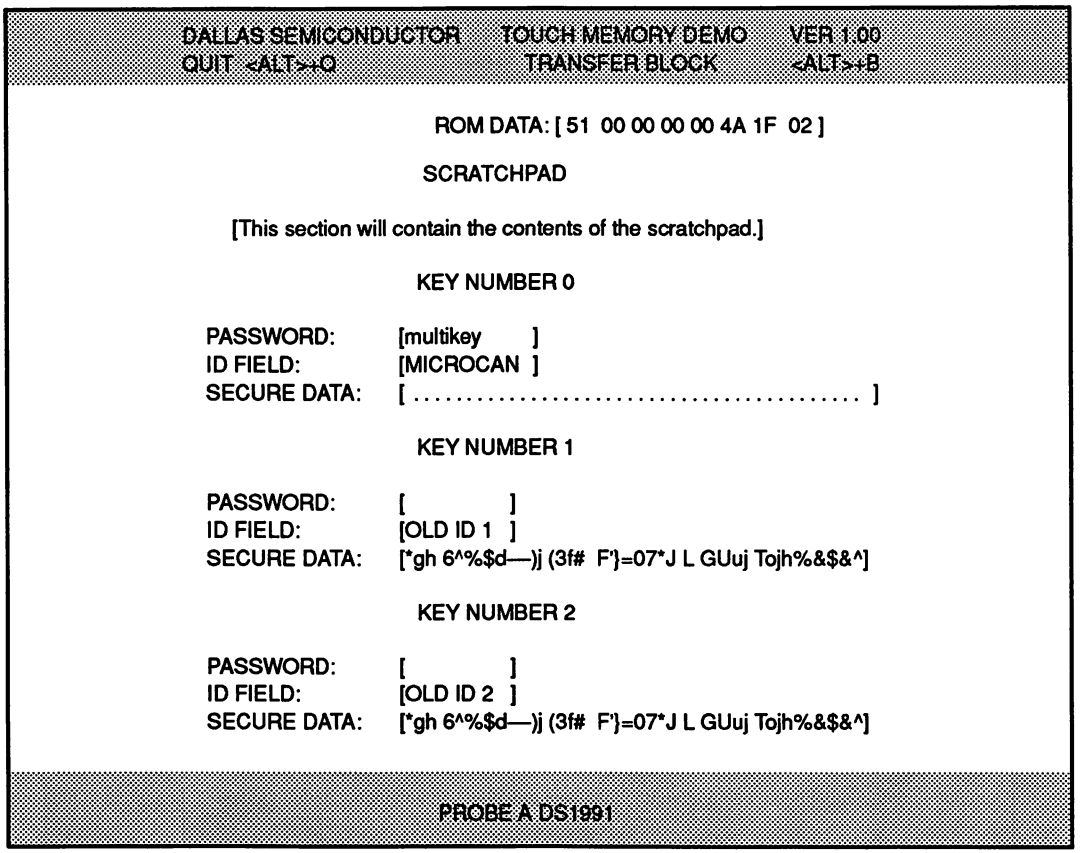
Step 1: To set or change a password, move the cursor to the desired subkey password field. Type in the new password (e.g., multikey) and press <ENTER>.

Notice that the random data in the secure field changes. This feature prevents someone from incrementally going through different passwords. Since the data changes for all variations of passwords, it is virtually impossible to tell if good or bad data is returned.

```
DALLAS SEMICONDUCTOR    TOUCH MEMORY DEMO    VER 1.00
QUIT <ALT>+Q                TRANSFER BLOCK       <ALT>+B

                    ROM DATA: [ 51  00 00 00 00 4A 1F  02 ]

                    SCRATCHPAD

        [This section will contain the contents of the scratchpad.]

                    KEY NUMBER 0

PASSWORD:       [multikey   ]
ID FIELD:       [OLD ID 0 ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    KEY NUMBER 1

PASSWORD:       [           ]
ID FIELD:       [OLD ID 1 ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    KEY NUMBER 2

PASSWORD:       [           ]
ID FIELD:       [OLD ID 2  ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    PROBE A DS1991
```

Step 2: Probe the DS1991. The random characters in the Secure Data field will change and the cursor will automatically move down to the ID field.

Step 3: Enter the new ID for this subkey (example: MICROCAN) and press <ENTER>.


DALLAS SEMICONDUCTOR TOUCH MEMORY DEMO VER 1.00
QUIT <ALT>+Q TRANSFER BLOCK <ALT>+B
ROM DATA: [51 00 00 00 00 4A 1F 02]
SCRATCHPAD
[This section will contain the contents of the scratchpad.]
KEY NUMBER 0
PASSWORD: [multikey]
ID FIELD: [MICROCAN]
SECURE DATA: [.....]
KEY NUMBER 1
PASSWORD: [ ]
ID FIELD: [OLD ID 1]
SECURE DATA: [*gh 6^%$d—)j (3f# F'}=07*J L GUuj Tojh%&$&^]
KEY NUMBER 2
PASSWORD: [ ]
ID FIELD: [OLD ID 2]
SECURE DATA: [*gh 6^%$d—)j (3f# F'}=07*J L GUuj Tojh%&$&^]
PROBE A DS1991

Step 4: Probe the DS1991. The secure data field will change to all periods. This indicates that the secure memory was erased and now new data can be entered. (example: "this is a secret ... SHHHH! ")

```
DALLAS SEMICONDUCTOR    TOUCH MEMORY DEMO    VER 1.00
QUIT <ALT>+Q                TRANSFER BLOCK       <ALT>+B

                    ROM DATA: [ 51  00 00 00 00 4A 1F  02 ]

                    SCRATCHPAD

[                                                        ]

                    KEY NUMBER 0

PASSWORD:       [multikey      ]
ID FIELD:       [MICROCAN  ]
SECURE DATA:    [this is a secret... SHHHH! ..................... ]

                    KEY NUMBER 1

PASSWORD:       [              ]
ID FIELD:       [OLD ID 1  ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    KEY NUMBER 2

PASSWORD:       [              ]
ID FIELD:       [OLD ID 2  ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                    PROBE A DS1991
```

2) Entering Secure Data

Step 1: Move the cursor to the desired subkey data field. Type in the new data (example: this is a secret... SHHHH!) and press <ENTER>.

Step 2: Probe the DS1991. The data is then loaded into the DS1991 and the screen is updated.

Step 3: The data can be modified in the secure data field as long as the password and the ID correctly match those in the DS1991.

3) Entering an Invalid Password

Step 1: Move the cursor back to the password field. Enter a new password and press <ENTER>.

Step 2: Probe the DS1991. The data in the secure data field will turn to random data. Step 1 and 2 can be repeated with different passwords and the random data will change. Re-enter the correct password using steps 1 and 2; the correct secure data will appear.

4) Transfer Block

This function enables the transfer of data, in blocks, from the scratchpad to any of the secure subkeys. The scratchpad is divided into eight 8–byte blocks, which can be transferred, individually or all together, into a subkey.

Step 1: Before a block can be transferred from the scratchpad to a subkey, the scratchpad must be set up with the ID field, the password, and the information to be transferred written in the scratchpad. Suppose the DS1991 had been initialized in the last example with a password in subkey 0 of "multikey" and ID field of MICROCAN. To set up the part for a block transfer, the scratchpad must be written to exactly as shown in the following steps.

```
DALLAS SEMICONDUCTOR    TOUCH MEMORY DEMO    VER 1.00
QUIT <ALT>+Q                TRANSFER BLOCK       <ALT>+B

                ROM DATA: [51 00 00 00 00 4A 1F 02]

                SCRATCHPAD

 [MICROCANmultikeyPut the information you want transferred here      ]

                KEY NUMBER 0

PASSWORD:       [multikey      ]
ID FIELD:       [MICROCAN      ]
SECURE DATA:    [this is a secret... SHHHH! ..................... ]

                KEY NUMBER 1

PASSWORD:       [              ]
ID FIELD:       [OLD ID 1      ]
SECURE DATA:    ['gh 6^%$d—)j (3f#  F'}=07'J L GUuj Tojh%&$&^]

                KEY NUMBER 2

PASSWORD:       [              ]
ID FIELD:       [OLD ID 2      ]
SECURE DATA:    ['gh 6^%$d—)j (3f#  F'}=07'J L GUuj Tojh%&$&^]
```

Step 2: Block Select – To select a block of data in the scratchpad to be transferred, enter <ALT>+B. The first eight bytes of the scratchpad will be highlighted. Using the right and left <arrow> keys, the block moves to the next or previous block. When the appropriate block is selected, press <ENTER>. If you want to transfer the entire scratchpad, move the cursor all the way to the right until the entire scratchpad is highlighted.

Step 3: Select Destination Subkey – Using the Up and Down <arrow> keys, move the cursor to the subkey desired (in our case, 0), and then press <ENTER>.

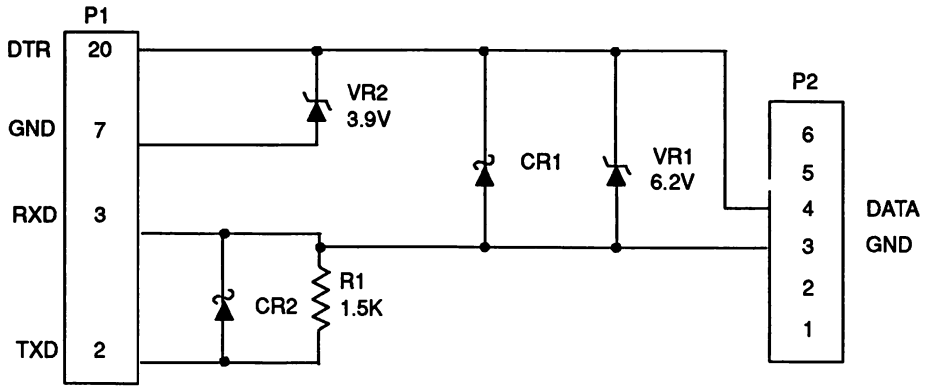
Step 4: Probe the DS1991 to complete the transfer. To confirm the transfer, the data in the selected blocks in the scratchpad will turn into nulls, shown by the program as periods.

```
DALLAS SEMICONDUCTOR    TOUCH MEMORY DEMO    VER 1.00
QUIT <ALT>+Q                TRANSFER BLOCK    <ALT>+B

                ROM DATA: [ 51 00 00 00 00 4A 1F 02 ]

                SCRATCHPAD

[ ..........................................................]

                KEY NUMBER 0

PASSWORD:       [multikey      ]
ID FIELD:       [MICROCAN  ]
SECURE DATA:    [Put the information you want transferred here      ]

                KEY NUMBER 1

PASSWORD:       [              ]
ID FIELD:       [OLD ID 1      ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]

                KEY NUMBER 2

PASSWORD:       [              ]
ID FIELD:       [OLD ID 2      ]
SECURE DATA:    [*gh 6^%$d—)j (3f#  F'}=07*J L GUuj Tojh%&$&^]
```

RS232 PORT

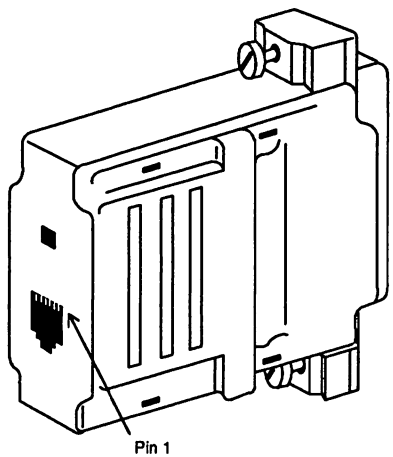
The serial communications port of a PC is easily converted to communicate with Touch Memories. The DS9097 Serial Port Adaptor in this kit combines the transmit and receive data lines of the computer's RS232C interface into a single bi-directional data line, known as 1-Wire. In addition, the adaptor circuitry reverses the polarity of the signal applied to the Touch Memory and limits the voltages to a range that is acceptable to the chip. Figure 2 shows a schematic of the DS9097 Serial Port Adaptor.

DS9097 SERIAL PORT ADAPTOR SCHEMATIC Figure 2



BILL OF MATERIALS

ITEM	REF DES.	DESCRIPTION
1	CR1, CR2	DIODE, ERA82-004 SCHOTTKY
2	VR1	DIODE, 6.2V ZENER
3	VR2	DIODE, 3.9V ZENER
4	P1	CONNECTOR, 25-PIN DB, FEMALE
5	P2	CONNECTOR, RJ-11, FEMALE
6	R1	RESISTOR, 1.5K OHM 1/4 WATT, 1% TOL.



The UART (typically an 8250) is set up with one start bit, six data bits, and one stop bit. With a baud rate of 115,200, each bit will be 8.68 µs long. With this configuration, each 6-bit word sent out by the UART combines to become a single bit on the 1-Wire. The start bit from the UART provides the falling edge and the data sync time described in the section, "Communicating with Touch Memories."

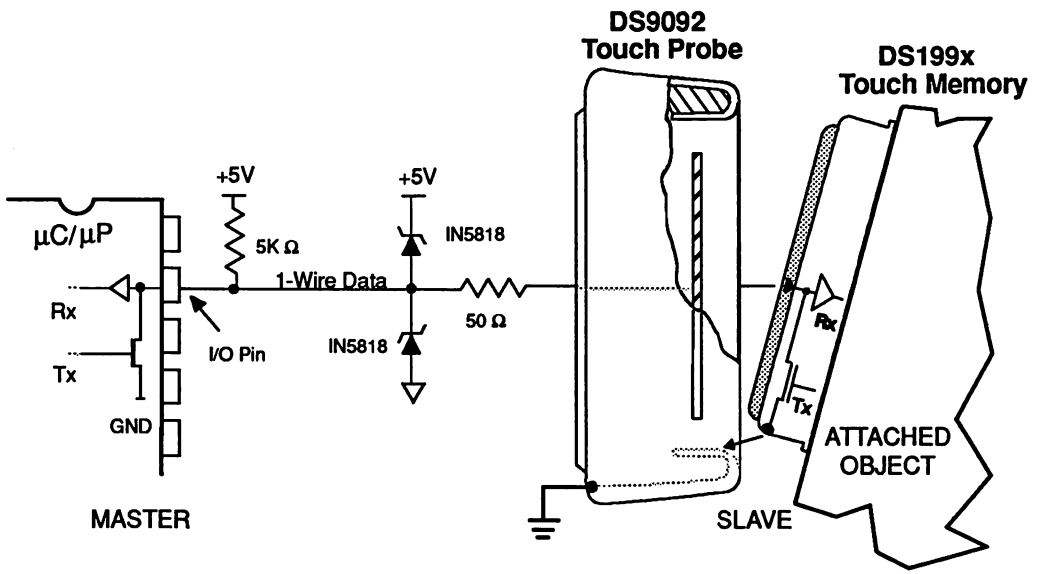
When a one is written on the 1-Wire, all six bits in the UART are set to one. When that word is transmitted, the waveform and timing on the 1-Wire conform to the sequence for a write one. This principle holds true for writing a zero as well as reading data.

For more detailed information on 1-Wire and its protocol, refer to Dallas Semiconductor Application Note #23, "Using the 1-Wire Protocol."

TROUBLESHOOTING GUIDE

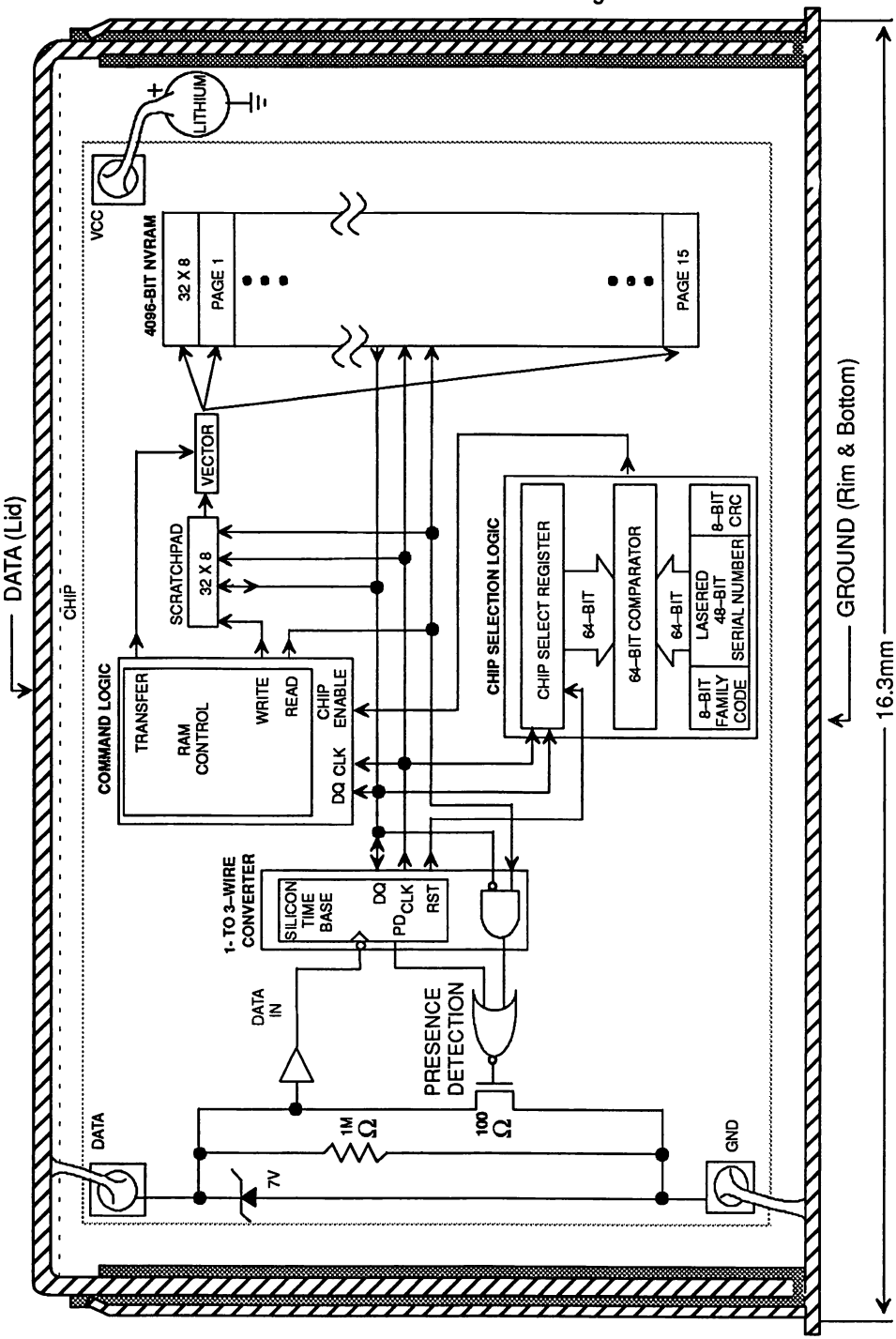
SYMPTOM	POSSIBLE CAUSE	CORRECTIVE ACTION
Software can't find PC adaptor.	PC adaptor attached to wrong port.	Enter the port # when starting the demo software, e.g., Touch 2 *<ENTER>*.
Software still can't find PC adaptor when using correct COM port.	PC COM port may not be able to operate at 115.2K baud.	Try demo on another PC.
Software finds PC adaptor, but won't read a Touch Memory.	Broken wire in Touch Memory Probe.	Run DIAGNOSE.EXE (UTIL DISK = Toolkit\)
Screen begins to display data and then stops and clears halfway through operation.	Probe removed too quickly.	Leave probe in contact with Touch Memory until screen refresh is complete.

TOUCH READER/WRITER Figure 3



Touch Memory facilitates chip-to-chip communication with just one signal wire. This signal wire plus ground results in forgiving alignment. The chip reads – as well as writes – data correctly even when the connection is momentary.

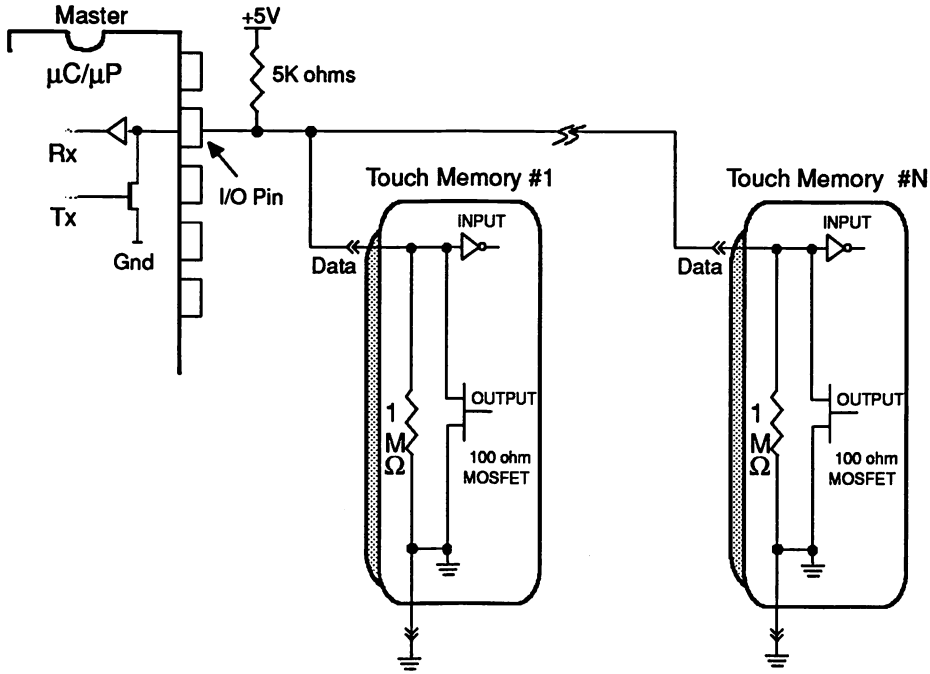
BLOCK DIAGRAM OF THE DS1993 TOUCH MEMORY Figure 4



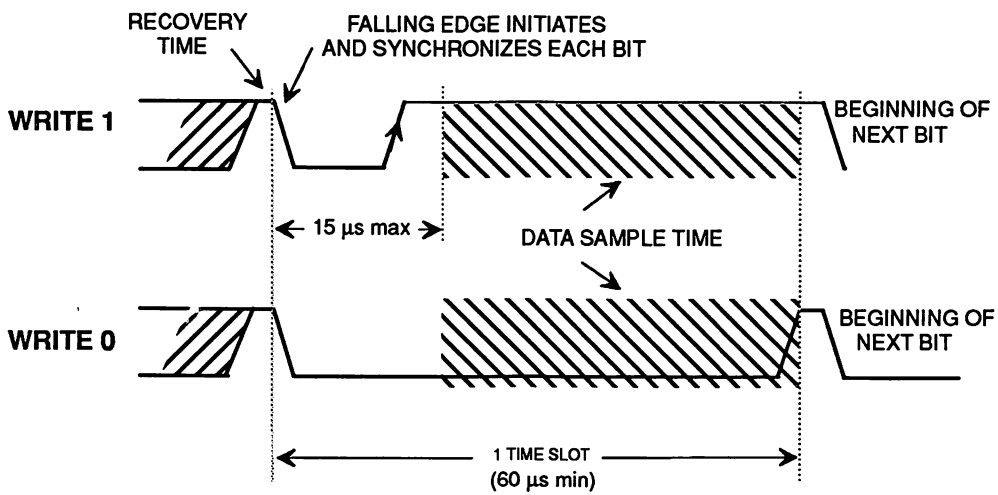
The Touch Memory Chip multiplexes address, data, and control onto a single bond pad that is extended directly to the lid of the MicroCan enclosure. Making the container and the electrical contacts one in the same achieves the greatest economy.

COMMUNICATING WITH TOUCH MEMORIES

- Communication between a master and multiple Touch Memories takes place over a single bidirectional data line.

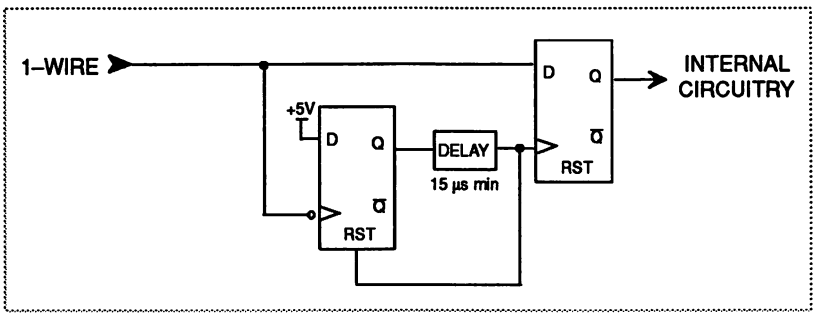


- The data line is passively pulled up with a 5K ohm resistor, but actively pulled down by the master or a Touch Memory.
- Data is transferred on the bus in discrete time intervals called time slots.
- 1's or 0's are represented with short or long active pull-down times within a time slot.

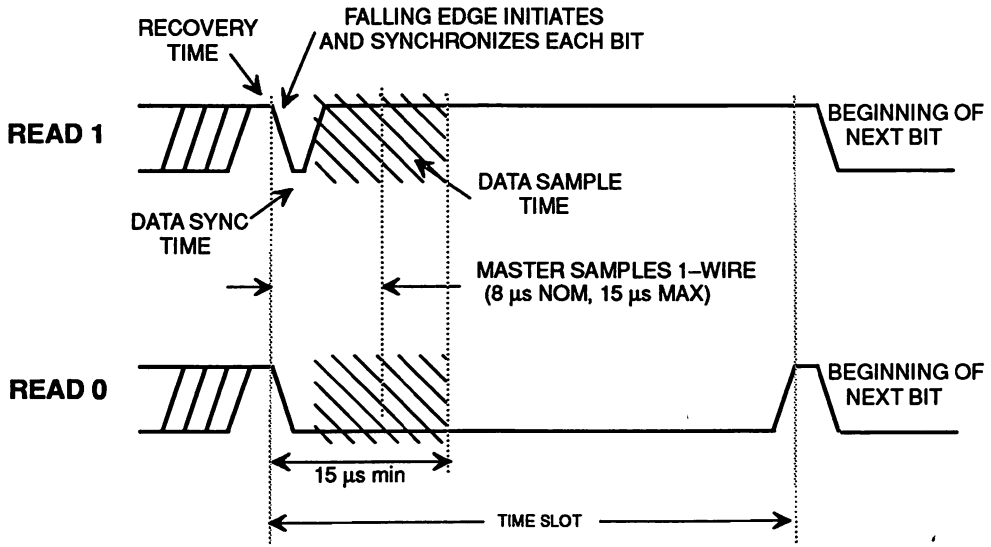


- To write a logic 1, the 1-Wire must return high within the data sample window. To write a logic 0, the 1-Wire must remain low for the entire data sample window.
- Synchronization is achieved on the falling edge of the time slot driven by the master. The master must hold the 1-Wire low for a minimum of 1µs.

EQUIVALENT CIRCUIT – WRITE MODE

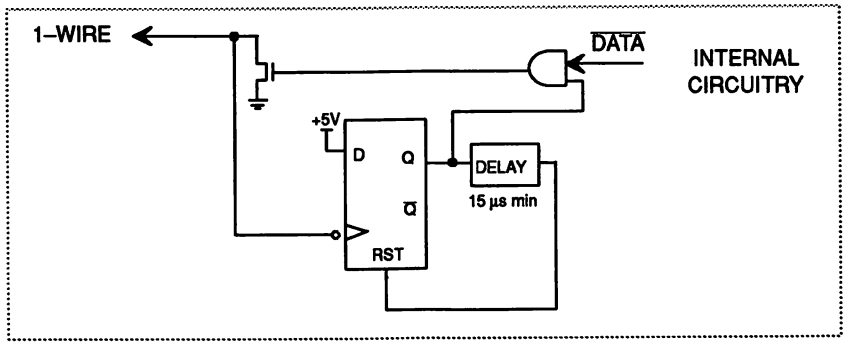


- On a read cycle, the ideal time for the master to sample the data line is 8 µs from the beginning of the time slot.
- The master drives the 1-Wire low during the data sync time (1 µs minimum) and then releases it. The Touch Memory then controls the state of the 1-Wire during the data sample time.

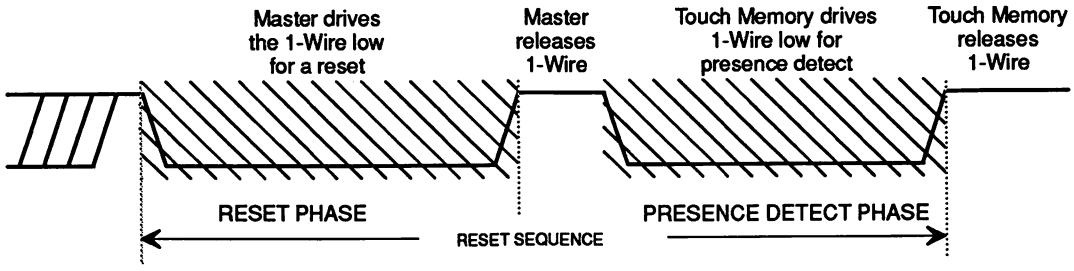


- The 1-Wire is passively pulled up 15 - 60 µs after the start of the time slot.

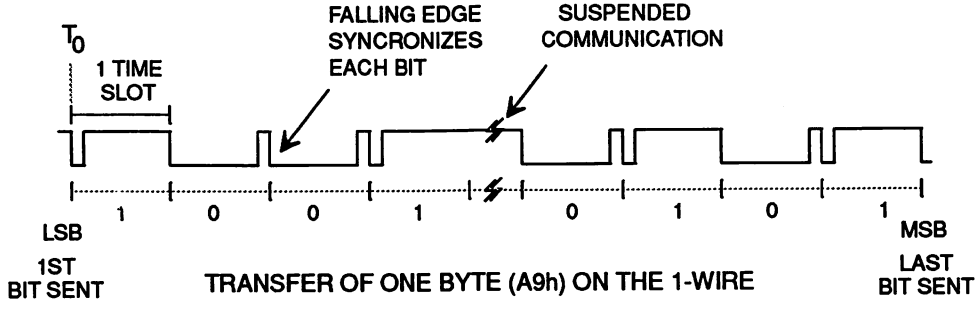
EQUIVALENT CIRCUIT – READ MODE



- The presence detect pulse notifies the master when a Touch Memory is first contacted.



- Communications can be suspended for any length of time between time slots with the 1-Wire left high.
- In all communications, the LSB is transmitted first .



SOME COMMON QUESTIONS AND ANSWERS ABOUT TOUCH MEMORY

1. What is a Touch Memory?

A Touch Memory is a reprogrammable, silicon label in a 16mm coin-shaped package. The Touch Memory attaches data to virtually any object for instant accessibility. Data is transferred quickly and reliably, with a momentary touch contact.

2. Why is the silicon chip in a coin-shaped package?

Touch Memory benefits from the high-volume, low-cost packaging methodology of canning. Called a MicroCan, the stainless steel enclosure preserves the chip even when subjected to corrosive conditions. Its bright, smooth surface easily wipes clean. In addition to serving as a protective housing, the MicroCan's metallic shell serves as the point of electrical contact – the chip communicates through the lid.

3. Why use a silicon chip as a label?

Silicon chip labels surpass ink-on-paper bar codes in alterability and greater data capacity. A silicon label holds up-to-date, relevant information because it can be reprogrammed while still attached. This portable data carrier can hold up to 100 times the information of a bar code.

4. Where does the term "Touch" come from?

Information is transferred to and from the chip with a simple touch. The MicroCan's curved edges guide a probe for self-alignment, much like positioning a cap on the rim of a bottle.

5. How is a Touch Memory affixed?

Like a post-it note, only in silicon, Touch Memory affixes by peel-and-stick adhesives. In addition, Dallas packages Touch Memory in a flanged MicroCan that can be snapped into a retainer for mechanical mounting.

6. What are the economics of Touch Memory?

The cheapest way to read or write a circuit is to extend a single bond pad to a larger external surface for contacting – the lid of the can in this case. Address, data, and control are multiplexed onto the bond pad. Communication with the other chips in a host system can be accomplished with the lowest overhead; chip-to-chip data transfers pass through inexpensive metal stampings.

7. How can the Touch Memory operate correctly when the contact is momentary and uncertain?

Two verification techniques ensure data integrity: a memory scratchpad and a CRC (Cyclical Redundancy Check). Data is first written to the scratchpad and then verified before it is transferred to memory. Once is transfer is initiated, a copy of the scratchpad is faithfully reproduced in memory, even if the contact is broken.

When data is written to the Touch Memory, the host calculates a CRC for each page of data. This information is appended to the data packet. When data is read, its integrity is validated by the CRC.

8. How fast can data be transferred?

Data is transferred bidirectionally using long and short pulses, similar to Morse Code, at speeds up to 16.6 Kbits/sec, making the transfer appear to be instantaneous.

9. How does the reader know that a Touch Memory is there?

Each member of the Touch Memory family announces its connection to a reader with a presence detect signal. With this feature, a host reader can remain in a low-power, standby state awaiting the contact of a Touch Memory.

10. What system resources are needed to read a Touch Memory?

A Touch Memory communicates using 5-volt CMOS logic signals and can be read or written using a spare microcontroller or microprocessor I/O pin. An existing hand-held reader can thus be easily modified to access a Touch Memory. Dallas Semiconductor provides a variety of mating receptacles that are compatible with the Touch family.

11. What features and memory capacities are available?

The read/write memory portion of the chip remains nonvolatile for 10 years. The MicroCan withstands over one million contacts. The Touch Memory's unique lasered-in serial number, which authenticates and registers the device, is permanent and cannot be altered. Memory sizes range from 64 to 4,096 bits.

12. How well does a Touch Memory stand up to harsh environments or outdoor exposure?

Because of its 305 stainless steel composition, the MicroCan shell is a durable package that resists corrosion from moisture, sterilizing solutions, and acidic compounds extremely well. This type of steel is commonly used for food and chemical processing. Mud and dirt can be easily cleaned off of the Touch Memory due to the MicroCan's smooth surface.

13. What are some typical applications?

Touch Memory can be used as an electronic key for access control. Due to its ruggedness and the low cost of the reader, the Touch Memory is an attractive alternative to commonly used magnetic stripe and bar-coded cards. Touch Memories can be affixed to existing photo-ID badges or snapped into the key fob accessory that Dallas offers.

Another application is manufacturing, where Touch Memory is attached to the workpiece to become a travelling record of operations performed. Touch Memory can be read or written while attached to the object, allowing records to be update as the manufactured goods flow down an assembly line. It can later serve as a maintenance record when the goods are used in the field.

Hands-on workers, such as hospital nurses, can collect data conveniently with portable instruments and subsequently dump the records into a host system with the same touch contact for error-free documentation.

Dallas Semiconductor Corporation Assumes No Responsibility for the use of Any Circuitry Other than Circuitry Embodied in a Dallas Semiconductor Product. No Other Circuit Patent Licenses Are Implied. Information Contained Herein Supersedes Previously Published Specifications on these Devices from Dallas Semiconductor.

DALLAS SEMICONDUCTOR CORPORATION
4401 SOUTH BELTWOOD PARKWAY, DALLAS, TEXAS 75244-3292
TEL: 214-450-0448
FAX: 214-450-0470

MAY 1992
TOUCH MEMORY STARTER KIT